IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Attorney Docket No.: **JA9-98-196**

In re Application of: §	
§	Examiner: **DELGADO, M.**
TADOKORO ET AL. §	
§	
Serial No.: **09/510,569** §	Art Unit: **2143**
§	
Filed: **22 FEBRUARY 2000** §	
§	
For: **HARDWARE SETUP METHOD** §	

RECEIVED

JUN 1 6 2003

Technology Center 2100

APPEAL BRIEF

MS Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Brief is submitted in triplicate in support of the Appeal in the above-identified application.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The present application is assigned to International Business Machines Corporation, the real party of interest.

RELATED APPEALS AND INTERFERENCES

No related appeal is presently pending.

STATUS OF THE CLAIMS

Claims 1-8 and 17-20 stand finally rejected by the Examiner as noted in the Final Office Action dated March 3, 2003 and the Advisory Action dated May 7, 2003.

STATUS OF AMENDMENTS

No amendment was submitted subsequent to the Office Action dated November 27, 2002.

SUMMARY OF THE INVENTION

Conventionally, hardware setup of a computer system is carried out by one of the following three ways:

(1)　the hardware setup is entirely performed by an operating system, and the computer system has no hardware setup program of its own;

(2)　the hardware setup is entirely performed by the computer system's own hardware setup program; or

(3)　the hardware setup program is performed by both the computer system's own hardware setup program and an operating system.

In (2) above, the hardware setup program can be recorded either on a portable recording medium, such as a floppy disk or CD-ROM, or on a fixed recording medium, such as a hard drive.

The installation of the hardware setup program requires some efforts. Such effort becomes significant for companies that have to install a large number of computer systems of the

same model in which the same hardware setting needs to be kept throughout all the computer systems. For example, in order to perform the hardware setup, the hardware setup program has to be installed in each individual computer system. In addition, the hardware setup program must be reinstalled after the hardware setup program has been upgraded, and the reinstallation process typically requires a tremendous amount of time when there are many computer systems needed to be set up. Consequently, it would be desirable to provide an improved method for performing hardware setup operations on a computer system to facilitate the above-mentioned hardware setup process.

In accordance with a preferred embodiment of the present invention, a hardware setup program and a set of dynamic link modules are stored in a server data processing system, as shown in server computer **22** of Figure **2**. Then, a data processing system is physically coupled to the server data processing system via a data processing system network such as a network **21** in Figure **2**, as depicted in block **S41** of Figure **4**. In response to a request to execute the hardware setup program by the data processing system, the hardware setup program is executed within the server data processing system, as shown in blocks **S42** and **S43** of Figure **4**. The hardware configuration data within the data processing system is subsequently modified according to instructions generated from the execution of the hardware setup program within the server data processing system, as depicted in block **S53** of Figure **5**. Finally, the data processing system is physically de-coupled from the server data processing system after a completion of the hardware setup operation. With the present invention, the hardware setup can be performed without installing the hardware setup program into the data processing system that requires the hardware setup operation.

ISSUE

Is the Examiner's rejection of Claims 1-8 and 17-20 under 35 U.S.C. § 102(e) as being anticipated by *Rakavy et al.* (US 6,324,644) well-founded?

GROUPING OF THE CLAIMS

For purposes of this Appeal, Claims 1-8 and 17-20 stand or fall together as a single group.

ARGUMENT

The Examiner's rejections of Claims 1-8 and 17-20 are not well-founded and should be reversed.

I. _Rakavy_ does not teach or suggest the claimed storing step

The purpose of the claimed invention is to perform a hardware setup operation on a data processing system without the hardware setup program being stored in the data processing system itself. Instead, both the hardware setup program and the associated dynamic link modules are stored in a server data processing system. Hence, Claim 1 (and similarly Claims 5 and 17) recites a step of "storing a hardware setup program and a plurality of dynamic link modules in a server data processing system" (lines 3-4).

On page 2 of the Final Office Action, the Examiner asserts that the claimed storing step is disclosed by _Rakavy_ in col. 10, lines 55-60. Specifically, col. 10, lines 55-60 of _Rakavy_ states:

> A loader services thread **426** provides the management workstation **200** application with an interface to the module loader **630**. The management workstation **200** can send messages to the module loader **630** containing code and data to be loaded into RAM, and executed as an additional transient network enhanced BIOS thread.

It is clear from the above-mentioned paragraph that _Rakavy_ does not teach or suggest the claimed storing step in col. 10, lines 55-60. Importantly, _Rakavy_ does not teach or suggest the hardware setup program and the dynamic link modules are stored in a server data processing system (which seems to be management workstation **200** in Figure **1** of _Rakavy_ according the Examiner's characterization).

The Examiner then asserts that "[t]o provide this information from floppy or hard disk..., it would have been obvious to some one of ordinary skill in the art that this information was already stored on the workstation (Figure 1, 200) prior to it being accessed." It is not clear what information the Examiner was referring to, but based on the context, Appellants assume the Examiner was referring to the hardware setup program and the dynamic link modules. However,

there is no basis to the Examiner's assertion that the hardware setup program and the dynamic link modules have already been stored in workstation **200**. Because the claimed invention includes novel features that are not taught or suggested by *Rakavy*, the § 102 rejection is improper.

II.   *Rakavy* does not teach or suggest the claimed coupling and de-coupling steps

Before performing a hardware setup operation, the data processing system is physically coupled to the server data processing system. Hence, Claim 1 (and similarly Claims 5 and 17) recites a step of "coupling a data processing system to said server data processing system via a data processing system network" (lines 5-6). After the hardware setup operation has been completed, the data processing system is physically de-coupled to the server data processing system. Hence, Claim 1 (and similarly Claims 5 and 17) recites a step of "de-coupling said data processing system from said server data processing system after a completion of said hardware setup operation" (lines 13-14). Even though the word "physically" was not explicitly recited in Claims 1, 7 and 17, the data processing system cannot be coupled to the server data processing system via any software means without first being physically coupled to the server data processing system; hence, the word "physically" is implied.

On page 2 of the Advisory Action, the Examiner asserts the claimed coupling step is disclosed in col. 9, lines 1-5 of *Rakavy*. Specifically, col. 9, lines 1-5 of *Rakavy* states:

> The network enhanced BIOS **600** now "simultaneously" starts to user the network communication. The detail of the network protocol stack operation is given later, this section gives an overview of the transactions taking place. The network enhanced BIOS **600** first issues **419** an alert packet on the network **300**, and prepares to open a data connection.

It is clear from the above-mentioned paragraph that *Rakavy* does not teach or suggest the claimed coupling step in col. 9, lines 1-5.

On page 2 of the Final Office Action, the Examiner asserts that the claimed de-coupling step is disclosed in col. 7, lines 15-25 of *Rakavy* as returning control to the remote computer. Specifically, col. 7, lines 15-25 of *Rakavy* states:

> If the installation check routine determines **439** that no such signature is present, the rest of POST routine **520** executes **413** and control is passed **414** to a bootstrapping routine supplied by the operating system. As described above, the bootstrapping routine loads **432** the operation system and passed **415** CPU control to it.
>
> If the signature indicating the start of the network enhanced BIOS **600** code is found then a relocation routine **416** copies the network enhanced BIOS **600** from the non-volatile memory **125** into RAM **120**, and CPU control is passed **417** to the initialization routine **610** of the network...

Again, it is clear from the above-mentioned paragraph that *Rakavy* does not teach or suggest the claimed de-coupling step in col. 7, lines 15-25. Since the claimed invention recites novel features that are not taught or suggested by *Rakavy*, the § 102 rejection is improper.

III.   *Rakavy* does not teach or suggest the claimed modifying step

Claim 1 (and similarly Claims 5 and 17) recites a step of "modifying hardware configuration data within said data processing system according to instructions generated from said execution of said hardware setup program within said server data processing system" (lines 10-12).

On page 3 of the Final Office Action, the Examiner asserts that the claimed modifying step is disclosed by *Rakavy* in col. 4, lines 25-35. Actually, it is clear that *Rakavy* does not teach or suggest the claimed modifying step in col. 4, lines 25-35 because *Rakavy*'s invention is directed to a method and system of communicating with a computer through a network prior to booting the computer's operating system or after operating system failure (col. 4, lines 18-21), and it is well-known to one skill in the art that a computer system is "inoperable" prior to the booting with the computer's operating system or after an operating system failure. Hence, it would be improbable, if not impossible, for a computer system to modify any hardware configuration data before the boot-up process has been completed or after an operating system

failure. Because the claimed invention includes novel features that are not taught or suggested by *Rakavy*, the § 102 rejection is improper.

CONCLUSION

For the reasons stated above, Appellants believe that the claimed invention clearly is patentably distinct over the cited references and that the rejections under 35 U.S.C. § 102 are not well-founded. Hence, Appellants respectfully urge the Board to reverse the Examiner's rejection.

Please charge the IBM Deposit Account **50-0563** in the amount of $320.00 for submission of a Brief in support of Appeal. No additional fee or extension of time is believed to be required; however, in the event an additional fee or extension of time is required, please charge that fee or extension of time requested to the IBM Deposit Account **50-0563**.

Respectfully submitted,

Antony P. Ng
Registration No. 43,427
BRACEWELL & PATTERSON, LLP
111 Congress Avenue, Suite 2300
Austin, Texas 78701
512.542.2134

ATTORNEY FOR APPELLANTS

APPENDIX

1   1.   A method for performing a hardware setup operation on a data processing system, said

2   method comprising:

3   storing a hardware setup program and a plurality of dynamic link modules in a

4   server data processing system;

5   coupling a data processing system to said server data processing system via a data

6   processing system network;

7   in response to a request to execute said hardware setup program by said data

8   processing system, executing said hardware setup program within said server data

9   processing system;

10   modifying hardware configuration data within said data processing system

11   according to instructions generated from said execution of said hardware setup program

12   within said server data processing system; and

13   de-coupling said data processing system from said server data processing system

14   after a completion of said hardware setup operation.

1   2.   The method of Claim 1, wherein when said hardware setup operation required by said data

2   processing system exists within an operating system running on said data processing system, said

3   hardware setup program performs said hardware setup operation by using a service provided by

4   said operating system.

1   3.   The method of Claim 1, wherein when said hardware setup operation required by said data

2   processing system does not exist within an operating system running on said data processing

3   system, said hardware setup program calls a BIOS program within said data processing system

4   to perform said hardware setup operation.

1   4.   The method of Claim 1, wherein said method further includes transferring one or more

2   of said dynamic link modules from said server data processing system to said data processing

3   system via said data processing system network as a result of said execution of said hardware

4   setup program.

1   5.   An apparatus for performing a hardware setup operation on a data processing system, said

2   apparatus comprising:

3   means for storing a hardware setup program and a plurality of dynamic link

4   modules in a server data processing system;

5   means for coupling a data processing system to said server data processing system

6   via a data processing system network;

7   means for executing said hardware setup program within said server data

8   processing system, in response to a request to execute said hardware setup program by

9   said data processing system;

10   means for modifying hardware configuration data within said data processing

11   system according to instructions generated from said execution of said hardware setup

12   program within said server data processing system; and

13   means for de-coupling said data processing system from said server data processing

14   system after a completion of said hardware setup operation.

1   6.   The apparatus of Claim 5, wherein when said hardware setup operation required by said

2   data processing system exists within an operating system running on said data processing system,

3   said hardware setup program performs said hardware setup operation by using a service provided

4   by said operating system.

1   7.   The apparatus of Claim 5, wherein when said hardware setup operation required by said

2   data processing system does not exist within an operating system running on said data processing

3   system, said hardware setup program calls a BIOS program within said data processing system

4   to perform said hardware setup operation.

1   8.   The apparatus of Claim 5, wherein said apparatus further includes means for transferring

2   one or more of said dynamic link modules from said server data processing system to said data

3   processing system via said data processing system network as a result of said execution of said

4   hardware setup program.

17.   A computer program product for performing a hardware setup operation on a data processing system, said computer program product comprising:

program code means for storing a hardware setup program and a plurality of dynamic link modules in a server data processing system;

program code mean for coupling a data processing system to said server data processing system via a data processing system network;

program code means for executing said hardware setup program within said server data processing system, in response to a request to execute said hardware setup program by said data processing system;

program code means for modifying hardware configuration data within said data processing system according to instructions generated from said execution of said hardware setup program within said server data processing system; and

program code means for de-coupling said data processing system from said server data processing system after a completion of said hardware setup operation.

18.   The computer program product of Claim 17, wherein when said hardware setup operation required by said data processing system does not exist within an operating system running on said data processing system, said hardware setup program calls a BIOS program within said data processing system to perform said hardware setup operation.

19.   The computer program product of Claim 17, wherein when said hardware setup operation required by said data processing system does not exist within an operating system running on said data processing system, said hardware setup program calls a BIOS program within said data processing system to perform said hardware setup operation.

1   20.   The computer program product of Claim 17, wherein said computer program product

2   further includes program code means for transferring one or more of said dynamic link modules

3   from said server data processing system to said data processing system via said data processing

4   system network as a result of said execution of said hardware setup program.